Recommender Systems

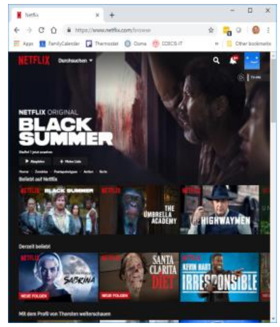
CS6780 – Advanced Machine Learning
Spring 2019

Thorsten Joachims
Cornell University

Reading:
Y. Koren, R. Bell, C. Volinsky, Matrix Factorization Techniques for Recommender Systems, IEEE Computer, 42:8, 2009. (link)

Movie Recommender

Recommendation
- Movie to watch



News Recommender

Recommendation
- Portfolio of newsarticles

Voice Assistant

Recommendation for "Alexa, play music"
- Playlist

Recommender Systems

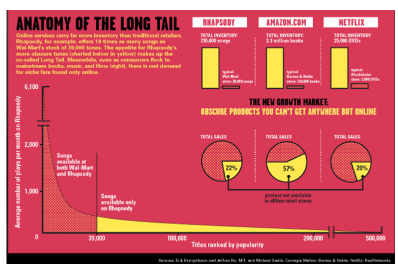
Examples
- Netflix: Movies
- Amazon: Products
- Spotify: Music
- YouTube: Videos
- Xbox Live: Games/Players
- Facebook: News

Problem
- There are far more "items" than an individual user could browse.

Goal
- Narrow down the choices to the items that are likely of interest to user.

The Long Tail

(Chris Anderson, 2004)



When do Recommender Systems work?

- Main Ideas
  - Past user preferences are predictive of future user preferences.
    - Example: If user u enjoyed action movies with Arnold Schwarzenegger in the past, recommend more action movies with Arnold Schwarzenegger.
  - There is a small number of user types.
    - Example: Users u_1 and u_2 both like the Red Hot Chilli Peppers. If u_1 also likes Linkin Park, then recommend Linkin Park to u_2.

Setup

- Set of users: U
- Set of items: V
- Ratings Y: $U \times V \rightarrow \Re$
  - Explicit Feedback
    - Star rating [1-5]
  - Implicit Feedback
    - Watched/skipped [0,1]
    - Visited web pages [1]

Observed Rating Matrix $\tilde{Y}$

$$\begin{array}{cc} & \text{Items} \\ \text{Users} & \begin{bmatrix} & 4 & 5 & & & & & & \\ 4 & & 3 & & & & & 2 & \\ 3 & 4 & & & & & & & \\ & 4 & 4 & & & 2 & & & \\ & & & 5 & 3 & & & & \\ & & & 4 & & 4 & & & \\ & & & & & & 5 & 3 & \\ & & & & 2 & & & 3 & 3 \\ 1 & & & & & & 4 & & 4 \end{bmatrix} \end{array}$$

Content-Based Recommendation

- Idea:

  Supervised learning for each row or column

  $h_u: X_v \rightarrow Y$

  $h_v: X_u \rightarrow Y$
- Challenge:

  Need to come up with features for users and/or items.

Observed Rating Matrix $\tilde{Y}$

$$\begin{array}{cc} & \text{Items V} \\ \text{Users U} & \begin{bmatrix} & 4 & 5 & & & & & & \\ 4 & & 3 & & & & & 2 & \\ 3 & 4 & & & & & & & \\ & 4 & 4 & & & 2 & & & \\ & & & 5 & 3 & & & & \\ & & & 4 & & 4 & & & \\ & & & & & & 5 & 3 & \\ & & & & 2 & & & 3 & 3 \\ 1 & & & & & & 4 & & 4 \end{bmatrix} \end{array}$$

Collaborative Recommendation

- Idea:

  Find users with similar ratings and fill in unobserved ratings.

  Find items with similar ratings and fill in unobserved ratings.

Observed Rating Matrix $\tilde{Y}$

$$\begin{array}{cc} & \text{Items V} \\ \text{Users U} & \begin{bmatrix} & 4 & 5 & & & & & & \\ 4 & & 3 & & & & & 2 & \\ 5 & 4 & 4 & & & & & & \\ & 4 & 4 & & & 2 & & & \\ & & & 5 & 3 & & & & \\ & & & 4 & & 4 & & & \\ & & & & & & 5 & 3 & \\ & & & & 2 & & & 3 & 3 \\ 1 & & & & & & 4 & & 4 \end{bmatrix} \end{array}$$

Matrix Completion Model

Observed Rating Matrix $\tilde{Y}$

$$\begin{array}{cc} & \text{Items V} \\ \text{Users U} & \begin{bmatrix} & 4 & 5 & & & & & & \\ 4 & & 3 & & & & & 2 & \\ 5 & 4 & 4 & & & & & & \\ & 4 & 4 & & & 2 & & & \\ & & & 5 & 3 & & & & \\ & & & 4 & & 4 & & & \\ & & & & & & 5 & 3 & \\ & & & & 2 & & & 3 & 3 \\ 1 & & & & & & 4 & & 4 \end{bmatrix} \end{array} = \overset{k}{\begin{bmatrix} \; \\ \; \\ \; \end{bmatrix}} \times \begin{bmatrix} \qquad\qquad \end{bmatrix} k$$

- Low rank assumption: rank k
- For each user u_i and item v_j

  $$Y_{ij} = u_i \, v_j$$
- Learn feature vectors u_i and v_j for each user/item

Matrix Completion Training

Observed Rating Matrix $\tilde{Y}$

$$\begin{array}{cc} & \text{Items V} \\ \text{Users U} & \begin{bmatrix} & 4 & 5 & & & & & & \\ 4 & & 3 & & & & & 2 & \\ 5 & 4 & 4 & & & & & & \\ & 4 & 4 & & & 2 & & & \\ & & & 5 & 3 & & & & \\ & & & 4 & & 4 & & & \\ & & & & & & 5 & 3 & \\ & & & & 2 & & & 3 & 3 \\ 1 & & & & & & 4 & & 4 \end{bmatrix} \end{array} = \overset{k}{\begin{bmatrix} \; \\ \; \\ \; \end{bmatrix}} \times \begin{bmatrix} \qquad\qquad \end{bmatrix} k$$

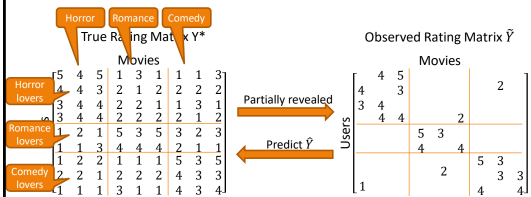
Given: Sample S of observed entries of $\tilde{R}$

Training: Solve for U and V with k rows/cols respectively

$$\min_{U,V} \sum_{(i,j) \in S} \left(\tilde{Y}_{ij} - u_i v_j\right)^2$$

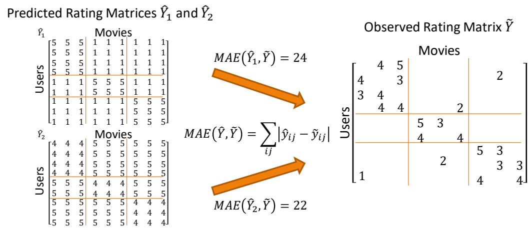
Prediction: Fill in entries not in S with $Y_{ij} = u_i \, v_j$

Movie Recommendation



→ Missing Not At Random (MNAR) Problem

MNAR and Evaluation



→ Severly biased performance estimates!

Why is the Data MNAR?

- User Induced MNAR

Why is the Data MNAR?

- System Induced MNAR